

Systém pro archivaci výkresové dokumentace

System for archiving drawings

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

Společnost MSV STUDÉNKA s.r.o. poskytla svůj archiv výkresové dokumentace dobrovolně, bezúplatně a za podmínky, že data z archivu nebudou distribuována s výslednou prací.

V Ostravě 2. května 2011

.....

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 2. května 2011

.....

Rád bych na tomto místě poděkoval všem, kteří mi s prací pomohli, protože bez nich by tato práce nevznikla.

Abstrakt

Tato bakalářská práce se zabývá tvorbou informačního systému pro archivaci výkresové dokumentace. Nejprve bude rozebrána problematika archivace dat se zaměřením na důvody a specifika archivace výkresové dokumentace. Dalším krokem bude návrh informačního systému, který bude navržen s ohledem na aspekty problematiky pokryté úvodním rozbořem. Poté bude část práce věnovaná výběru vhodných technologií jako aplikačního frameworku a posléze celé aplikační platformy. V závěru bude nastíněno několik možností budoucího rozšíření celého systému.

Klíčová slova: výkresová dokumentace, archivace, informační systém, webová aplikace, php, Nette Framework

Abstract

This bachelor thesis deals with creation of an information system for archiving drawings. First part will discuss issues of archiving data focusing on reasons and specific needs of drawings archiving. Next step will cover design of information system which will be designed with regard to all aspects of the issues covered in the first part. The next part of this work will be dedicated to choosing appropriate technologies such as application framework and then the whole application platform. The final part will outline several options of future extension of the system.

Keywords: drawings, archiving, information system, web application, php, Nette Framework

Seznam použitých zkratk a symbolů

ACL	– Access Control List
AD	– Active Directory
AJAJ	– Asynchronous JavaScript and JSON
AJAX	– Asynchronous JavaScript and XML
CSRF	– Cross Site Request Forgery
DRY	– Don't Repeat Yourself
DVD	– Digital Versatile Disc
DWG	– drawing
HTML	– HyperText Markup Language
IIS	– Internet Information Services
KISS	– Keep It Short and Simple
LDAP	– Lightweight Directory Access Protocol
MVC	– Model View Controller
PHP	– PHP: Hypertext Preprocessor
XSS	– Cross Site Scripting

Obsah

1	Úvod	5
2	Problémy archivace a návrh řešení	6
2.1	Velikost archivačního prostoru	6
2.2	Řízení přístupu	6
2.3	Ochrana před poškozením	7
2.4	Ochrana před odcizením	7
2.5	Rychlost nalezení hledané informace	7
2.6	Předání dokumentace obchodním partnerům	8
2.7	Finanční náročnost	8
3	Digitalizace	9
3.1	Překreslení do CAD systému	9
3.2	Využití digitizéru	9
3.3	Skenování	9
3.4	Použité řešení	10
4	Analýza IS	12
4.1	Funkční požadavky	12
4.2	Nefunkční požadavky	12
4.3	Konceptuální model řešení	12
4.4	Analýza procesů	15
5	Výběr technologií IS	18
5.1	Adresářová služba	18
5.2	Jazyk a příslušné frameworky	18
5.3	Databázové systémy	19
5.4	Operační systém	20
5.5	Využití technologií v IS	21
6	Implementace IS	22
6.1	Autentikace a autorizace uživatelů	22
6.2	Vyhledávání dat v archivu	22
6.3	Optimalizace přenášených dat	23
6.4	Uchovávání a získávání dat	24
6.5	Hromadné nahrávání a analýza dat	25
7	Další rozvoj IS	27
7.1	Administrovatelné regulární výrazy pro identifikaci souborů	27
7.2	OCR	27
7.3	Podpora více datových skladů	27
7.4	Analýzátor využití jednotlivých souborů	27
7.5	Identifikace verze DWG	28

7.6	Bezpečnější přihlašování	28
7.7	Implementace NTLM	28
7.8	Virtuální systém	28
8	Závěr	29
9	Reference	30

Seznam tabulek

1	Porovnání velikostí dat	23
---	-----------------------------------	----

Seznam obrázků

1	ER diagram pro databázové schéma	11
---	--------------------------------------------	----

1 Úvod

V prostředí malých a středních podniků působících v oblasti strojírenství dochází k problémům s archivací a přístupem k výkresové dokumentaci nutné pro výrobu. Velmi viditelná je tato situace u společností v odvětví kolejových vozidel. Se spoluprací souhlasila společnost MSV STUDÉNKA s.r.o., která má v této oblasti několikaleté zkušenosti, vlastní výrobní zařízení, vytváří novou výkresovou dokumentaci a disponuje archivem starší výkresové dokumentace.

Nová dokumentace již vzniká v programech jako AutoCAD, SolidWorks, ZwCAD, Catia a dalších podobných, ovšem převážná část výrobní dokumentace vznikala v letech minulých, kdy tento software nebyl dostupný. Vzhledem k tomu, že tato dokumentace je často v obrovských formátech na papíře s vysokou gramáží, je uchovávání už několika výkresů velmi prostorově náročné. V případě společnosti MSV STUDÉNKA zabírá dokumentace značnou část technické kanceláře a jeden sklad. Tento způsob uchování přináší několik problémů:

- velikost archivačního prostoru
- řízení přístupu
- ochrana před poškozením ať už samotným používáním nebo některou z živelných katastrof
- ochrana před odcizením
- rychlost nalezení hledané informace
- předání dokumentace obchodním partnerům
- finanční náročnost řešení

Protože i samotná společnost si výše zmiňované problémy uvědomuje, začala již svůj archiv digitalizovat. Převést papírovou dokumentaci do formátů zpracovatelných konstrukčním softwarem by bylo nejen časově a finančně náročné, ale pro její následné použití i zbytečné. Dokud se neprovádí změna ve výkresu samotném, postačuje naskenování výkresu do bezeztrátových formátů jako TIFF nebo PDF v poměru 1:1.

Řízení přístupu k datům a autentizace uživatelů, jako nedílná součást řešení, musí být navržena s ohledem na existující firemní infrastrukturu.

Současná situace tak velmi nahrává vytvoření softwarového informačního systému, který by archivaci zajistil a vyřešil většinu nastíněných problémů.

2 Problémy archivace a návrh řešení

Problémy, které byli nastíněny v úvodu, bude třeba rozebrat a proto je každému z nich věnována samostatná podsekce.

2.1 Velikost archivačního prostoru

Přestože jeden papírový výkres zabírá zanedbatelné množství prostoru, představíme-li si 100000 takovýchto výkresů, už se o zanedbatelný prostor nejedná. Ba co víc. Vezmeme-li modelový případ 100000 výkresů (A4 - 40000ks, A3 - 30000ks, A2 - 20000ks, A1 - 7000ks, A0 - 3000ks), tak pro takovýto archiv je zapotřebí prostor o velikosti 20x10m, což je 200m² [3].

2.1.1 Řešení

Řešením je digitalizace. Při průměrné velikosti souborů (A4 - 70 kB, A3 - 120 kB, A2 - 200 kB, A1 - 350 kB, A0 - 600 kB) bude pro modelový případ stačit 14GB dat, což jsou 3 disky DVD [3]. Tento objem je při kapacitě dnešních úložných zařízení velmi snadno zvládnutelný včetně možnosti geograficky oddělených záloh, popřípadě zálohování do cloudu¹. Jednotlivé možnosti digitalizace budou rozebrány v následující kapitole.

2.2 Řízení přístupu

V prostředí společností s „papírovým archivem“ je řízení přístupu velmi problematické. Pokud je toto řízení řešeno pouhým klíčem od dveří do archivu, popřípadě od uzamykatelné skříně, ve které je dokumentace umístěna, existuje riziko, že bude klíč zkopírován a přístup získají další osoby.

Při větším množství osob s oprávněním vstupu do archivu je problém s udržitelností takového systému. Monitorování přístupu samotného je také problematické, protože odemknutí zámku nemusí nikdo povolit a ani o něm nemusí nikdo vědět.

2.2.1 Řešení

Řešení je zavedení elektronického řízení přístupu. Ať už pomocí čipových karet nebo zadáním jména a hesla v elektronickém archivu. Elektronické řízení může přinést nové možnosti jako například:

- logování² přístupu
- možnost časového omezení přístupu, např. jen během pracovní doby
- jednodušší přidávání nových oprávněných osob

¹<http://open-tube.com/what-is-cloud-backup-a-beginners-guide-to-cloud-backup>

²Zaznamenávání událostí

Informační systém, který bude vytvořen bude implementovat ověřování vůči interní databázi uživatelů a pro možnost univerzálního nasazení ve firemním prostředí bude implementována podpora pro ověřování vůči systému LDAP.

2.3 Ochrana před poškozením

Jak už bylo řečeno v úvodu, jednotlivé výkresy byly historicky vytvářeny na papír. Tato metoda uchování je vhodná pro jednorázové vytvoření výkresu, ale při používání dochází ke značným deformacím. Hrozí roztržení, ušpinění a ztráta výkresu samotného. Osoby neznalé práce s výkresovou dokumentací ji mohou také nadměrně poškozovat nesprávnou manipulací a překládáním. Postupem času také dochází ke korozi a degradaci papíru [4], což velmi ztěžuje další práci s dokumentací. Zálohování takovéto dokumentace pro případ rizika požáru nebo živelné katastrofy je extrémně problematické.

2.3.1 Řešení

Rozumným řešením, s ohledem na zachování možnosti práce s dokumentací, je digitalizace. Jednotlivé možnosti digitalizace budou rozebrány v následující kapitole.

Zároveň se s digitalizací naskytne možnost efektivního zálohování a to včetně předtím nemyslitelného geograficky odděleného zálohování, pro případ požáru nebo živelných katastrof.

2.4 Ochrana před odcizením

Protože výkresová dokumentace je jeden z klíčových prvků pro výrobu, a tedy konkurenční výhodou, je třeba pamatovat na možnost odcizení. Vytvořené zabezpečení ovšem musí brát v potaz, že s dokumentací je třeba i po zabezpečení pracovat jak v konstrukční kanceláři, tak je třeba nějakým způsobem předávat jednotlivé výkresy do výroby.

2.4.1 Řešení

V případě fyzického uložení výkresové dokumentace v archivu je řešení pořízení kamerového systému pro prostory archivu. Monitorování pohybu výkresu v rámci prostor společnosti už se jeví jako velmi problematické. V případě elektronického uložení lze přesně monitorovat přístup k jednotlivým výkresům, ovšem vynesení výkresu v digitální podobě ze společnosti budeme předcházet velmi těžko.

2.5 Rychlost nalezení hledané informace

Při práci s výkresovou dokumentací se nejčastěji pracuje s číslem výkresu, které je pro něj unikátní a až poté s jeho názvem. Rychlost nalezení konkrétního výkresu je velmi důležitý aspekt práce s archivem výkresů. Za situace fyzického uložení výkresů trvá nalezení jednoho konkrétního kusu minuty až hodiny. V případě, že je navíc výkres momentálně ve výrobě, nebude možné jej v archivu najít a jeho dohledání tak bude v

řádech desítek hodin až dnů. Pokud se s výkresem pracovalo v poslední době, je jeho nalezení otázkou minut, avšak stále je třeba člověk, který má v systému uložení přehled.

2.5.1 Řešení

Řešením se jeví vytvoření indexu, který by stál nad všemi uloženými výkresy. Vzhledem k omezenému množství atributů nutných pro nalezení výkresu, tedy jeho unikátní číslo a případně název, by bylo velmi vhodné, aby vyhledávání v indexu bylo co nejvíce optimalizováno na tyto dva atributy.

2.6 Předání dokumentace obchodním partnerům

V případě, že objednatel žádá vyrobení dílu nebo je poptávána výroba dílu, ke kterému nemá daná výrobní společnost příslušnou dokumentaci, je tato dodána z archivu společnosti, jež výrobu požaduje. Dochází tak nejen k nutnému zveřejnění vlastnictví konkrétního výkresu, ale také hrozí riziko poškození nebo ztráty.

2.6.1 Řešení

Se zveřejněním vlastnictví konkrétního výkresu nepůjde nic udělat, ale lze naprosto eliminovat možnost poškození nebo ztráty, pokud by se předávala pouze kopie daného výkresu. V případě fyzického uložení lze předat pouze ofocenou kopii, což je časově a finančně náročné. V případě elektronického uložení už lze předat jen soubor s minimálními časovými a finančními náklady.

Pokud společnost vlastní konkrétní výkres ve formátu použitelném pro přípravu výroby, může takovéto elektronické předání dokumentace i výrazně snížit cenu zakázky, jelikož odpadne část přípravy pro výrobu.

2.7 Finanční náročnost

Velmi výrazným problémem archivace dat je finanční náročnost na zabezpečení celého systému. V případě fyzického uložení je nutné počítat se zajištěním místnosti protipožárním systémem, bezpečnostním systémem a klimatizačním systémem. V případě elektronického uložení dat je nutné počítat s náklady na zálohování, licence a provoz systému.

2.7.1 Řešení

Jak ve fázi analýzy, tak při samotné implementaci systému bude brána v potaz podmínka na přiměřenou finanční náročnost. V maximální možné míře budou využívána otevřená řešení bez zatížení licencemi a systém bude navržen tak, aby přílišně nezatěžoval žádnou jemu poskytnutou část a bylo tak možné jej provozovat i na starším hardwaru a v pomalejších sítích.

3 Digitalizace

Digitalizace je převod z fyzické „analogové“ podoby do digitální tvaru. V případě této práce se bude jednat o převod papírové podoby výkresové dokumentace do její digitální reprezentace. Metod provedení je několik, ovšem každá s jinou výslednou užžitnou hodnotou a adekvátně tomu finanční a časovou náročností. Při výběru vhodné metody je tedy velmi nutné zvážit, jakými technickými a finančními prostředky společnost disponuje a zda přinesou vynaložené prostředky patřičný užitek.

3.1 Překreslení do CAD systému

Jedná se o klasický a nejčastěji používaný systém digitalizace výkresové dokumentace, při kterém je výkres položen vedle počítače jako předloha a následně je ručně překres. Přesněji lze říci, že je výkres znovu vykonstruován.

Tato metoda poskytuje nejlepší možné výsledky, kdy je výstupem formát DWG³ a nebo obdobný⁴.

Evidentní nevýhodou této metody je však časová a z toho plynoucí finanční náročnost. I u jednodušších konstrukčních dílů může technický pracovník strávit hodiny tvorbou nového výkresu. U složitějších konstrukcí to pak můžou být desítky hodin. Digitalizace jednoho výkresu pak může znamenat značné finanční nároky a efektivita pracovníka provádějícího digitalizaci je velmi nízká.

Tuto metodu digitalizace je vhodné zvolit až ve chvíli, kdy je potřeba výkresovou dokumentaci upravovat nebo s ní pracovat v softwaru využívajícím soubory DWG a podobné.

3.2 Využití digitizéru

Digitalizace digitizérem je obdobou překreslení do CAD systému, s tím rozdílem, že je nutné mít digitizér příslušného rozměru, což představuje zvláště u větších formátů, zejména A1 a A0, značnou finanční zátěž. Vlastní proces pak probíhá tak, že je výkres položen na digitizér, zkalibrován a jsou "odsňímány" jednotlivé konce čar pro získání příslušných souřadnic bodů, které jsou následně převedeny do elektronické podoby a přeneseny do CAD prostředí.

Tato metoda poskytuje přesné výsledky, ovšem je časově náročná a vyžaduje nezanedbatelné množství finančních prostředků, nutných k pořízení digitizéru příslušné velikosti.

3.3 Skenování

Skenování výkresové dokumentace je velmi rozšířená metoda digitalizace. Výstupem je rastrový obraz ve vysoké kvalitě, se kterým je možné dobře pracovat a případně jej později zdigitalizovat jednou z výše uvedených metod. Velkou výhodou je skutečnost,

³Binární formát pro uchování dvojrozměrných a trojrozměrných dat

⁴Někteří výrobci CAD softwaru používají kromě DWG také své vlastní proprietární binární formáty

že společnost nemusí nutně vlastnit skenovací zařízení příslušného formátu, které stojí značné finanční prostředky, ale může využít zařízení jiné společnosti nebo grafického studia. Skenování také přináší nezanedbatelnou úsporu času, protože zdigitalizování jednoho výkresu touto metodou trvá jen několik minut a to včetně zavedení výkresu včetně souboru do evidenčního systému.

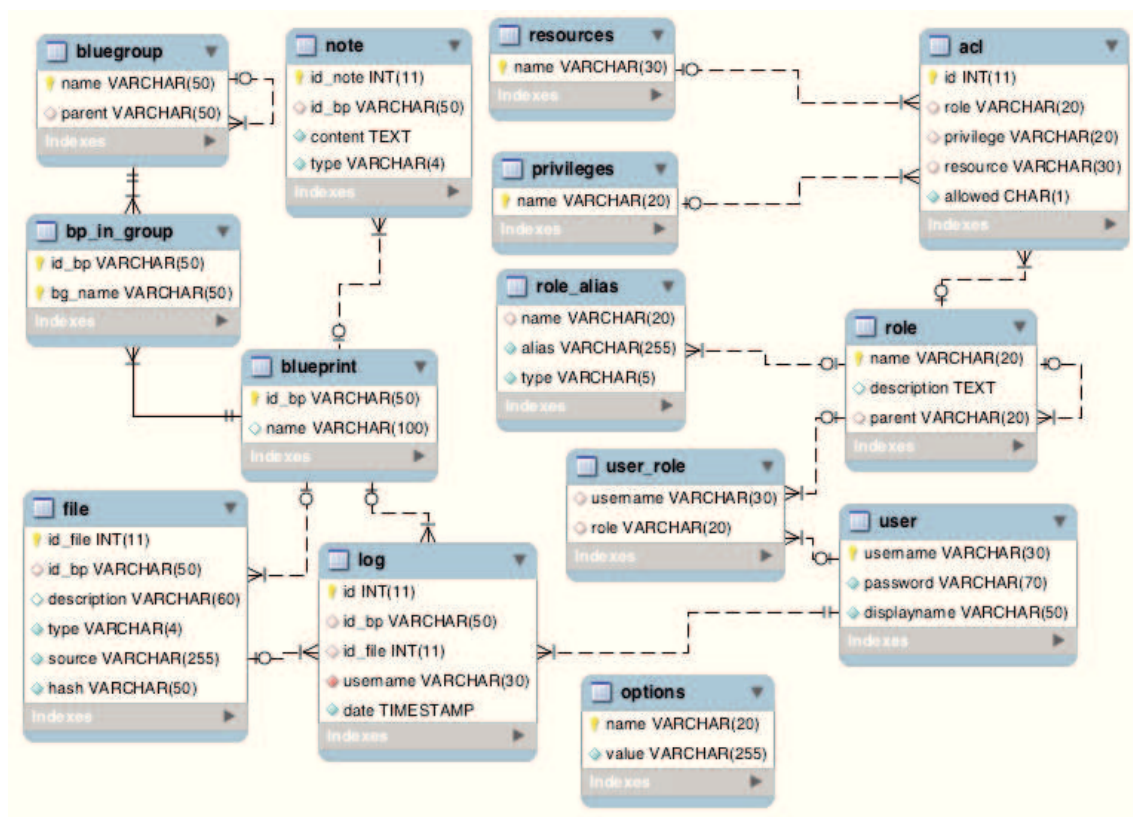
3.4 Použité řešení

V případě společností s velkým archivem a malým podílem „živých⁵ výkresů“ je digitalizace do formátů DWG a podobných velmi nákladná. Samotné naskenování výkresů a jejich následná archivace je nejen časově výrazně méně náročná, ale také finančně méně náročná.

Po konzultaci s konstruktérem společnosti MSV STUDÉNKA bylo zjištěno, že digitalizace metodou skenování a překreslování již ve společnosti probíhá, avšak velká část dokumentace se používá pouze v naskenované podobě a jen velmi malou část je nutné digitalizovat do DWG. Aktuálně má společnost zdigitalizováno přibližně 20000 výkresů, které tvoří 12GB dat.

Výsledný systém by tedy neměl výkresy v DWG nutně vyžadovat, ale neměl by jejich vložení omezovat.

⁵Jako živé jsou označovány výkresy, se kterými se aktivně pracuje.



Obrázek 1: ER diagram pro databázové schéma

4 Analýza IS

Potřebujeme evidenci výkresové dokumentace pro jednotný přístup uživatelů k digitalizovanému archivu včetně specifikace přístupových práv jednotlivých uživatelů.

4.1 Funkční požadavky

V rámci výrobního a obchodního procesu v oblasti strojírenského průmyslu je třeba nahlížet do velkého množství výkresové dokumentace. Jakmile je dokumentace vytvořena, jsou v ní prováděny pouze ojedinělé změny.

Mnoho uživatelů potřebuje ke své práci výkresovou dokumentaci určité úrovně přístupu, proto se počítá i s tímto bezpečnostním opatřením.

Potřebujeme evidovat informace o jednotlivých výkresech. Výkresy mohou být organizovány do skupin, například pro kompletní podvozkové ústrojí. K jednotlivým výkresům mohou být uchovávány poznámky.

Uživatelé získávají účet do systému s příslušností do skupin uživatelů, které je následně opravňují k určitým činnostem. Jednotliví uživatelé mohou být organizováni do skupin.

Skupiny výkresů představují výkresy, které mají vazbu. Např. se používají do stejného podvozkového ústrojí nebo je vytvořila stejná společnost.

K systému přistupují role s definovaným oprávněním:

- Uživatel - má právo vyhledávat v archivu
- Konstruktor - má práva jako Uživatel, a navíc může vkládat nová data a měnit data už vložená
- Administrátor - má právo zakládat nové uživatele, nastavovat oprávnění a měnit nastavení ovlivňující chování systému

4.2 Nefunkční požadavky

Výsledný systém poběží jako webová aplikace s uživatelským prostředím. Velmi důležitá je odezva prostředí pro uživatele a rychlost dosažení hledané informace. Role s oprávněním ke vkládání výkresové dokumentace musí mít možnost jednoduchým způsobem do systému vkládat více výkresu najednou.

4.3 Konceptuální model řešení

ER diagram je zobrazen na obrázku 1

4.3.1 Entity a jejich atributy

4.3.1.1 Výkres Entita Výkres reprezentuje výkres z výkresové dokumentace. Je u ní třeba evidovat následující atributy:

- číslo výkresu, které je unikátním identifikátorem
- název výkresu

4.3.1.2 Skupina Výkresů Entita Skupina Výkresů reprezentuje jeden a více výkresů pod společným názvem. Je u ni třeba evidovat následující atributy:

- název, který je unikátním identifikátorem
- nadřazenou skupinu, která je nepovinným atributem

4.3.1.3 Soubor Entita Soubor reprezentuje fyzický soubor digitalizované výkresové dokumentace náležící konkrétnímu výkresu. Je u ni třeba evidovat následující atributy:

- číslo soubor, které je unikátním identifikátorem
- číslo výkresy, ke kterém soubor náleží
- typ, který určuje jakého souborového typu je daný soubor
- popis, který obsahuje údaje, jež nemusí být ze souboru zřejmé
- zdroj, který je vlastním názvem souboru fyzicky uloženém v souborovém systému
- hash, který dovoluje následně kontrolovat unikátnost a integritu souboru

4.3.1.4 Poznámka Entita Poznámka reprezentuje poznámku náležící konkrétnímu výkresu. Je u ni třeba evidovat následující atributy:

- číslo poznámky, které je unikátním identifikátorem
- číslo výkresu, ke kterému poznámka náleží
- typ, pro identifikování druhu obsahu
- obsah poznámky

4.3.1.5 Uživatel Entita Uživatel reprezentuje uživatele přistupujícího k systému. Je u ni třeba evidovat následující atributy:

- uživatelské jméno, které je unikátním identifikátorem
- heslo
- zobrazované jméno

4.3.1.6 Role Entita Role reprezentuje roli v systému, které je možné přidělovat oprávnění a do kterých jsou uživatelé organizováni. Je u ni třeba evidovat následující atributy:

- název role, který je unikátním identifikátorem
- popis role, pro lepší přehlednost
- rodiče role, který je nepovinný

4.3.1.7 Alias role Entita Alias role představuje reprezentaci výše zmiňované role v jiných systémech. Je u ni třeba evidovat následující atributy:

- název role
- alias, který je vyjádřením role z jiného systému
- typ, který určuje, o jaký jiný systém se jedná

4.3.1.8 Právo Entita Práva reprezentuje možná práva, která lze využívat v rámci řízení přístupu k jednotlivým funkcionalitám systému. Je u ni třeba evidovat následující atributy:

- název práva, které je unikátním identifikátorem

4.3.1.9 Zdroj Entita Zdroj reprezentuje zdroj v systému, ke kterému je v rámci řízení přístupu možné přiřazovat práva. Je u ni třeba evidovat následující atributy:

- název práva, které je unikátním identifikátorem

4.3.1.10 ACL Entita ACL představuje určující prvek, který roli dovoluje nebo zakazuje využívat pro daný zdroj dané oprávnění. Je u ni třeba evidovat následující atributy:

- číslo pravidla, které je unikátním identifikátorem
- role
- zdroj
- oprávnění
- příznak povolení nebo odepření

4.3.1.11 Log Entita Log představuje záznam přístupu uživatele k výkresům a souborům v čase. Je u ni třeba evidovat následující atributy:

- číslo záznamu, které je unikátním identifikátorem
- číslo výkresu
- číslo souboru, které je nepovinné
- uživatelské jméno
- čas, kdy k události došlo

4.3.2 Relace mezi entitami

Z důvodu různorodosti uchovávaných formátů může jeden Výkres zahrnovat více Souborů. Mezi těmito entitami tedy bude vazba 1:N. K jednomu výkresu také může být vytvořeno více Poznámek a tedy i zde je vazba 1:N.

Výkres může být organizován v mnoha Skupinách Výkresů a stejně tak Skupina Výkresů může mnoho Výkresů obsahovat. Jedná se tedy o vazbu M:N.

Záznam v ACL se váže vždy k jednomu Uživateli, Zdroji a Právu. Všechny vazby jsou tedy 1:N.

Role může mít několik Aliasů z různých systémů a proto jde o vazbu 1:N.

Uživatel může náležet do několika rolí, stejně jako role mohou mít několik uživatelů. Jde tedy o vazbu 1:N.

Záznam do Logu se váže k jednomu uživateli, jednomu výkresu a může se vázat i k jednomu souboru. Jde tedy o vazby 0-1:N.

4.4 Analýza procesů

Mimo CRUD⁶ operace, které se vyskytují napříč systémem a které nemá smysl podrobně rozebírat, je třeba v systému podchytit tyto klíčové procesy:

1. Práce s výkresu
 - Vyhledání výkresu
 - Stažení výkresu
 - Zaznamenání přístupu
2. Řízení přístupu
 - Přihlášení uživatele
 - Autorizace uživatele
 - Odhlášení uživatele

⁶<http://www.infoworld.com/d/developer-world/rest-and-crud-impedance-mismatch-927>

3. Práce se soubory

- Analýza souboru
- Zařazení souboru do systému
- Kontrola integrity souboru

4.4.1 Podrobný popis procesů

4.4.1.1 Vyhledání výkresu Tento proces je nejčastější ze všech procesů vyskytujících se v systému. Uživatel specifikuje obecný řetězec, na základě kterého bude požadovat rychlé nalezení přesného výsledku. Vzhledem k povaze výkresové dokumentace a omezenosti atributů jednotlivých výkresů se uživatel bude nejčastěji dotazovat pomocí čísla výkresu nebo jeho názvu, popřípadě kombinace obojího.

4.4.1.2 Stažení výkresu Pokud bude výkres stahován ze systému, je třeba zajistit, aby bylo toto stažení zaznamenáno a soubor musí být stažen v korektním a použitelném stavu. Stažení tedy musí nutně předcházet kontrola integrity souboru pomocí kontrolního součtu a přejmenování výsledného souboru podle definovaných pravidel a s příponou souboru umožňující jeho snadné otevření asociovaným programem.

4.4.1.3 Zaznamenání přístupu Tento proces zajišťuje pasivní bezpečnost systému pro možnou budoucí analýzu úniku dat, popřípadě vyhodnocování využití výkresové dokumentace pro možnosti optimalizace.

4.4.1.4 Přihlášení uživatele Tento proces zajistí, že uživatel přistupující k systému bude ověřen vůči specifikovaným databázím uživatelů a následně mu bude povolen nebo odmítnut přístup do systému.

4.4.1.5 Autorizace uživatele Proces zajišťuje, že je v případě potřeby pomocí ACL ověřeno, že přihlášený uživatel má dovoleno využívat daného zdroje.

4.4.1.6 Odhlášení uživatele V tomto procesu bude zajištěno, že systém zruší aktivní přihlášení uživatele a nebude možno jej znovu použít.

4.4.1.7 Analýza souboru Proces zajistí, že bude uživateli usnadněno vkládání souborů do systému, v této fázi rozbořením názvu souboru.

4.4.1.8 Zařazení souboru do systému Proces zajistí, že nově vkládaný soubor, v případě neexistence výkresu, záznam o výkresu vytvoří a následně vytvoří záznam o souboru a soubor uloží do systému. V případě existence výkresu pak zajistí, že soubor již není vložen, a případně soubor vloží. Při vkládání také proces vytvoří kontrolní součet souboru.

4.4.1.9 Kontrola integrity souboru Proces bude zajišťovat, že soubor uložený v systému je shodný se souborem, který byl do systému vložen a tedy, že nedošlo k jeho poškození nebo modifikaci.

5 Výběr technologií IS

Protože při návrhu tohoto systému je brán ohled na cenu implementace a dalšího provozu, je výběr vhodných technologií velmi důležitý.

5.1 Adresářová služba

Provozovat adresářovou službu v infrastruktuře, kam bude systém nasazován, není pro funkčnost podmínkou. Vzhledem k tomu, že využití takovéto adresářové služby není jednoúčelové, bude určitě dobré se před nasazením systému zamyslet, zda spolu s ním nenasadit i adresářovou službu a tak nezjednodušit autentikaci uživatelů v celé síti společnosti.

5.1.1 LDAP

LDAP je zkratka pro Lightweight Directory Access Protocol, což je aplikační protokol pro dotazování a modifikaci adresářových služeb nad TCP/IP. V 80tých letech vznikla skupina standardů X.500 (DAP, DSP, DISP, DOP), které pokrývají adresářové služby. Zjednodušením standardu X.500 a zaměřením na TCP/IP vznikl LDAP [5].

5.1.2 Active Directory

Active Directory je implementace LDAP, spolu s dalšími technologiemi, společností Microsoft pro použití v prostředí systému Microsoft Windows. Pokud by společnost již využívala Windows Server, bylo by možné zavést použití Active Directory, pokud tak již společnost neudělala.

5.2 Jazyk a příslušné frameworky

Při výběru jazyka pro implementaci informačního systému jsem, vzhledem ke svým znalostem, zvažoval mezi PHP s Nette Frameworkem, Pythonem s frameworkem Django a C# s .NET frameworkem.

C# byl první neúspěšný kandidát, protože výsledný systém by byl vázán na využití IIS a Windows Serveru, což s sebou přináší další finanční nároky, které jsem si dal za cíl omezit. Také má pouze částečná znalost tvorby ASP.NET aplikací by byla pro systém limitující, což by zcela jistě vedlo k podcenění bezpečnosti výsledného řešení. Výhodou by však nepochybně byl vyšší výkon, který ovšem v prostředí malých a středních společností není natolik kritický faktor.

Python se ukázal jako velmi schopný jazyk, ovšem opět má nedostatečná znalost schopností Django frameworku by byla pro řešení limitující. Samotný Python však sehrál při vývoji systému velmi zásadní roli, protože díky němu mohly vzniknout první prototypy analyzátorů souborů a indexery existujícího archivu.

PHP Jako jazyk pro implementaci bylo tedy zvoleno PHP s Nette Frameworkem. S PHP mám největší zkušenosti a tak je tato volba rozumná. Nette Framework byl zvolen jednak proto, že se jedná o ryze český PHP framework a existuje tedy velká česká komunita⁷, která je schopna pomoci při vývoji a také při následném rozvoji systému i dalším osobám, ale také proto, že se vývoj frameworku už od počátku zaměřuje hlavně na bezpečnost a využití technologií a principů jako AJAX, AJAJ, DRY, KISS a velmi důležité MVC, které je v tomto frameworku nahrazeno MVP⁸. Framework má také velmi dobře implementovanou práci s formuláři, jejich validaci a další zpracování. Z pohledu bezpečnosti je framework odolný vůči útokům XSS, CSRF, url attack, control codes, invalid UTF-8, session hijacking, session stealing a session fixation⁹.

5.3 Databázové systémy

Spolu s Nette Frameworkem je využita i knihovna pro vytvoření abstraktní databázové vrstvy dibi¹⁰. Tato knihovna usnadní přenositelnost systému mezi různými databázovými systémy. Také umožní objektově pracovat s výsledky databázových dotazů a objektově tyto dotazy vytvářet.

Mezi databázovými systémy jsem zvažoval použití PostgreSQL, SQLite a MySQL. Databázová vrstva dibi podporuje všechny tyto systémy.

PostgreSQL jsem nepoužil, protože není součástí klasických LAMP¹¹ instalací. Ani poskytovatelé webhostingu příliš PostgreSQL neposkytují. Pokud by tedy došlo k nasazení systému, kde se administrátor zároveň stará i o webové stránky společnosti, je velmi malá šance, že bude mít povědomí i o PostgreSQL. Mé omezené znalosti PostgreSQL také limitovaly vývoj a nasazení systému.

SQLite jsem zvažoval, jelikož jeho použití je, vzhledem k uložení celé databáze do jediného souboru, velmi jednoduché. Záloha databáze by také byla mnohem jednodušší než u konkurentů. Ovšem jako limitující se ukázala podpora cizích klíčů, která byla do SQLite přidává až ve verzi 3.6.19¹², která je poměrně nová a nemusí se tedy běžně v systémech vyskytovat. K použití tohoto databázového systému nedošlo, jelikož s absencí cizích klíčů bylo využití SQLite shledáno jako nevyhovující.

MySQL bylo vybráno jako nejvhodnější varianta z několika důvodů:

- vysoká rozšířenost - součástí standardních instalací LAMP
- podpora cizích klíčů - engine InnoDB

⁷<http://nette.org/cs/komunita>

⁸<http://zdrojak.root.cz/clanky/nette-framework-mvc--mvp/>

⁹<http://nette.org/cs/hlavni-prednosti/bezpecnost>

¹⁰<http://phpfashion.com/dibi-pokrokovy-databazovy-layer>

¹¹Linux Apache MySQL PHP/Python

¹²<http://www.sqlite.org/foreignkeys.html>

- možnost přechodu na placené verze¹³ s podporou tvůrce databáze, společnosti Oracle
- předchozí zkušenost s nazením a správou

5.4 Operační systém

Z operačních systémů byly zvažovány systémy Microsoft Windows Server 2003 (nebo 2008, včetně R2 variant) verze Standard, a Ubuntu Linux.

Windows Server Předností Windows Server mimo jiné je, že umožňuje společnosti provozovat Active Directory, které by tak dovolilo využívat adresářových služeb zmiňovaných výše. Součástí Windows Server je také možnost zálohování a to jak plné, tak přírůstkové. Kromě Core verze Windows Serveru obsahují všechny verze IIS, což by společnosti zajistilo možnost provozovat webové aplikace a systémy využívající .NET Framework.

Windows Server také umožňuje provoz Microsoft SQL Server, tedy databázového systému společnosti Microsoft, se kterým umí pracovat databázová vrstva dibi a bude tedy možné systém provozovat i na tomto databázovém systému. Z hlediska možností instalací serverových aplikací není Windows Server nijak omezen oproti desktopovým systémům Windows. Pro administratory bez zkušeností s linuxovými systémy je také správa Windows Serveru jednodušší.

Pro malé a střední společnosti se jeví velmi vhodné nasazení Windows Small Business Serveru, který v sobě kromě základního Windows Serveru zahrnuje i Microsoft Exchange Server a zmiňovaný SQL Server.

Nevýhodou nasazení Windows Server je cenová náročnost, která v sobě zahrnuje jednak jednorázové licenční poplatky za samotný systém a dále poplatky za přístupové licence, tzv. CAL¹⁴, pro každého přistupujícího uživatele.

Pokud tedy společnost již nemá nasazeno Windows Server prostředí, mohlo by být jeho pořízení nákladnou investicí.

Ubuntu Linux Je nejoblíbenější linuxovou distribucí současnosti¹⁵. Systém jako takový vychází z distribuce GNU/Linux Debian, je vyvíjen společností Canonical a nadací Ubuntu Foundation a je poskytován pro komerční i nekomerční použití zdarma. Pro budoucího administrátora tak nebude problém testovat systém na různém hardwaru bez rizika porušování licenčního ujednání, bude moci provozovat na svém vlastním počítači desktopovou variantu systému Ubuntu a v případě potřeby bude moci využít rady od široké uživatelské komunity systému nebo si zaplatit podporu přímo od společnosti Canonical.

Na Ubuntu lze provozovat všechny software náležící do balíku LAMP, tedy Apache jako webový server, MySQL a PHP, což jsou technologie, které využívá i vytvářený

¹³<http://www.mysql.com/products/>

¹⁴Client Access License

¹⁵dle <http://distrowatch.com/>

systém. Lze také provozovat adresářovou službu LDAP, která centralizuje databázi uživatelů v celé společnosti.

Nevýhodou je složitější administrace, jelikož na většinu věcí neexistují grafické nástroje a je třeba pracovat v příkazové řádce a editovat textové soubory.

5.5 Využití technologií v IS

Společnost MSV STUDÉNKA s.r.o. disponuje jak Windows Serverem s Active Directory, tak serverem s Ubuntu Linuxem. Vzhledem k tomu, že ne všechny společnosti by byly ochotné investovat do Windows Serveru, a taky může nastat situace, že samotný Windows Server již kompletně využívá hardwarové prostředky a provoz dalšího systému by mohl způsobit pokles výkonu služeb běžících na Windows Serveru, bude systém vyvíjen tak, aby byl schopen pracovat na Windows Serveru, ale také Ubuntu Linux. Při vývoj tak bude třeba dbát na specifika tvorby aplikací v PHP pro tyto dva operační systémy. MySQL nijak nasazení omezovat nebude.

Bude také implementována podpora přihlašování přes Active Directory, která tak velmi usnadní začlenění systému do firemní infrastruktury a umožní jeho rychlejší nasazení do pracovního procesu.

6 Implementace IS

Během implementace systému byly vzaty v potaz všechny provedené analýzy a konzultace s pracovníky. Zde je uvedeno několik zajímavých částí vytvářeného systému a problematika, kterou řeší.

6.1 Autentikace a autorizace uživatelů

Přestože má společnost implementováno přihlašování přes Active Directory, nebylo by dobré se spoléhat pouze na tuto cestu. Problém nastane už při zavádění systému, kdy do něho nebude mít nikdo přístup a bude třeba ručně zasáhnout do databáze, což by mohlo způsobit nežádoucí nekonzistenci. Další problém může nastat ve chvíli, kdy uživateli vyprší platnost hesla a Active Directory jej tak začne automaticky odmítat. Z toho důvodu byla v systému vytvořena vlastní databáze uživatelů.

Pro zachování možnost využití přihlašování přes Active Directory bylo třeba vytvořit mapování, které by dovolilo jednoduchým způsobem systému říci, zda má uživatel přístup do systému a do které role náleží. K tomu bylo využito toho, že Active Directory umožňuje vyjádřit členství v roli jako řetězec. Byla tedy vytvořena mapovací tabulka, která roli z databáze systému přiřazuje roli z Active Directory vyjádřenou řetězcem. Tato nově vzniklá mapovací tabulka tam může do budoucna umožňovat zapojení libovolných systémů pro autentikaci a autorizaci uživatelů.

Pokud si při přihlašování uživatel zvolí možnost ověřování přes Active Directory, systém se dotáže doménového řadiče, zda uživatel zadal existující login a k němu správné heslo a poté si vyžádá informace o členství uživatele ve skupinách. Pokud není žádná ze skupin, jejichž je uživatel členem, nalezena v mapovací tabulce rolí, je uživateli zamítnut přístup. Zde vzniká omezení, které systém má. Role uživatele v Active Directory systém neprochází rekurzivně k rodiči, ale bere v potaz pouze poslední potomky. Pokusí-li se uživatel v roli „Technologové“, která je členem role „Přístup k archivu“, a ta je zahrnuta v mapovací tabulce, přihlásit k systému, bude mu přístup zamítnut, jelikož systém o něm získá pouze informaci o členství v roli „Technologové“, pro kterou nemusí mít mapovací záznam. Řešení tohoto nedostatku bude zcela jistě předmětem dalšího rozvoje systému.

Pokud je uživatel do systému přihlášen, jsou načteny role systému, do který náleží a které určují příslušné oprávnění uživatele. Nette Framework umožňuje ověřovat zda uživatel náleží do příslušné roli nebo zda má uživatel skrze některou ze svých rolí patřičné oprávnění k příslušnému zdroji. Ve výsledném systému je implemenována podpora pro oba tyto přístupy, a tak v případě rozšiřování systému je možné přidat do nových modulů kontrolu oprávnění, popřípadě zjemnit kontrolu oprávnění v existujících modulech.

6.2 Vyhledávání dat v archivu

Při prvotních analýzách byl kladen důraz na rychlost získání požadované informace. Skupině zaměstnanců byly předloženy dva návrhy možného vyhledávání dat. První poskytoval možnost specifikace hledaných dat, jako například formát čísla hledaného výkresu nebo typ hledaného souboru. Druhý obsahoval pouze jedno textové vyhledávací

Typ	Normal	Gzip
HTML	5,6 kB	1,9 kB
XML	2,3 kB	412 B
JSON	1,7 kB	397 B

Tabulka 1: Porovnání velikostí dat

pole, podobně jako má třeba Google¹⁶. Skupina si ihned vybrala druhou variantu, protože jejich nejčastější dotaz na archiv je, že potřebují libovolný soubor obsahující výkres a přitom znají celé číslo výkresu nebo jeho název a potřebují konkrétní výkres, a nebo znají část čísla výkresu a potřebují si vzpomenout na jeho název.

Vyhledávání dat tedy bylo do maximální možné míry zjednodušeno a zůstalo pouze jedno pole pro zadávání hledaného výkresu. Zadaný dotaz je poté rozložen na jednotlivá slova a ty jsou v databázi hledána jak v atributu číslo výkresu, tak jeho názvu. Vzhledem k tomu, že nad oběma atributy je v databázi vytvořen index, je prohledávání i většího množství dat velmi rychlé. Výsledek je poté uživateli zobrazen rozdělený na výkresy samotné a skupiny výkresů.

Protože hledaný dotaz se může často opakovat a v databázi nedochází často ke změnám, nabízí se vhodná příležitost pro využití cache. Nette Framework má integrovanou podporu pro práci s cache a umožňuje využívat různé způsoby uložení cachovaných dat, počínaje uložením každého ukládaného záznamu serializovaně do souborového systému a nebo uložením za pomoci modulu memcached¹⁷, který data udržuje v paměti RAM a nabízí k nim tak velmi rychlý přístup. Problémem využití memcached modulu je nutnost jeho dodatečné instalace a konfigurace. Pro vytvářený systém byla cache využita ve dvou fázích. V první fázi vývoje bylo jako úložiště využito DummyStorage, což je kostra využitelná pro tvorbu dalších úložišť zahrnutá v Nette Frameworku. DummyStorage má metody implementované tak, že korektně přijme data, ale poté se tváří, že žádná nemá. Velmi se tak hodí pro otestování funkčnosti a rychlosti systému, před nasazením ostré cache. V druhé fázi tvorby systému pak byla nasazena cache ukládající data do souborového systému, jelikož je takováto cache snadno přenositelná a zjednodušuje nasazení, protože nevyžaduje instalaci dalších modulů. Přestane-li cache využívaný souborový systém výkonostně dostačovat, lze ji s minimálním úsilím vyměnit na cache využívající právě modul memcached.

6.3 Optimalizace přenášených dat

Jelikož lze očekávat, že k systému nebudou přistupovat pouze uživatelé připojení rychlou LAN sítí, ale také přes pomalé bezdrátové nebo VPN spojení, bylo nutné systém vytvořit tak, aby přenášel minimum dat mezi serverem a uživatelem.

Technologie AJAX, která je běžně pro načítání dat v prostředí webových aplikací používána, z titulu X ve svém názvu využívá převážně formát XML. Formát XML je do

¹⁶www.google.com

¹⁷<http://www.memcached.org/>

značné míry dobře čitelný formát, se kterým lze následně velmi dobře pracovat, avšak jeho nevýhodou je velikost přenášených dat, což je problém, který je právě předmětem řešení.

Nette Framework má integrovanou podporu pro AJAX, tedy AJAX, kde je XML nahrazeno formátem JSON¹⁸, který je oproti XML úspornější na velikost přenášených dat a je velmi snadné s ním pracovat v JavaScriptu na straně klienta.

Samotné prvotní stažení HTML stránky znamená značný objem dat, který změnou formátu nepůjde eliminovat. Jako velmi vhodné řešení se jeví nasazení komprese, konkrétně Gzip, která je v prostředí webových aplikací velmi rozšířená. I použitý Nette Framework obsahuje podporu pro kompletní kompresi výstupu. Komprese Gzip byla tedy zapnuta a všechny generované HTML stránky jsou zkomprimované.

Srovnání velikostí přenášených dat je v tabulce 1.

6.4 Uchovávání a získávání dat

Při samotné implementaci systému bylo nutné zvážit, jak vkládané soubory uchovávat. Jako možné řešení se ukázalo uchovávání v databázi a uchovávání v souborovém systému.

Uchovávání v databázi by systému umožnilo mít jak binární data, tak informace co data obsahují na jednom místě. Zjednodušilo by se také zálohování, kdy by se zálohovalo pouze jedno uložení. Problémem takového řešení je značná velikost databáze a tím pádem doba, po kterou je databáze odstavena, než proběhne záloha. Pokud by navíc byla databáze umístěna na jiném serveru než aplikační server, na kterém bude spuštěn informační systém, může získávání souboru ovlivnit rychlost získání jiných dat, jelikož propustnost síťového rozhraní, kterým bude druhý server s největší pravděpodobností připojen, bude přenos limitovat.

Uchovávání v souborovém systému sice přináší vyšší režii k získání jednoho souboru, protože je nutné nejprve v databázi najít informace o souboru a umístění a až poté přistoupit k souboru v rámci souborového systému, avšak přináší jiné, nezanedbatelné výhody. Mezi ně patří v první řadě zkrácení odstávky funkčnosti celého řešení. Pokud databáze nebude obsahovat binární data, bude její velikost řádově menší, oproti stavu, kdy by je obsahovala. Poté bude zálohování databáze rychlejší a může být znovu rychle uvedena do provozu. Následovat bude zálohování datového skladu souborů, které, při zavedení opatření zajišťujících konzistenci souborového systému, může probíhat za běhu systému. Pro uživatele tak bude zaznamatelná odstávka systému naprosto minimální.

Další výhodou, které takovýto způsob uchovávání přinese, je možnost využití dalšího serveru jako datového skladu a výsledného poskytování souboru přímo z tohoto serveru bez zatěžování aplikačního serveru. V obzvláště velkých nasazeních pak může být databáze v jednom místě a datové sklady replikovány do několika

¹⁸<http://www.json.org/>

míst, což umožní při požadavku uživatele vybrat datový sklad, který je právě jemu nejbližší nebo ke kterému má nejlepší připojení.

Ve výsledném systému bylo zvolena metoda uchovávání v souborovém systému s uložením informací o souborech v databázi.

Po nahrání souboru do systému je pro něj vytvořen kontrolní součet pomocí algoritmu SHA1 a dále je stejným algoritmem vytvořen unikátní název souboru. Kontrolní součet slouží pro pozdější ověření, zda je soubor stále ve stavu, v jakém byl do systému vložen a zda nedošlo k manipulaci s tímto souborem nebo zda nebyl soubor poškozen. Unikátní název slouží k uložení souboru v souborovém systému. Soubor je ukládán pouze pod tímto názvem, bez přípony určující jeho typ. Typ, spolu s kontrolním součtem, je uložen v databázi, což systému umožňuje prověřený soubor uživateli poskytnout se srozumitelným názvem a příponou, která v jeho operačním systému umožní snadné otevření souboru příslušným programem.

Při testování se ukázalo jako dosti problematické uchovávání tisíců souborů v jediném adresáři a proto jsou ve výsledném systému soubory uchovávány v podadresářích s názvem složeným z prvních dvou znaků názvu souboru. Toto zanoření je možné provést i vícekrát, ovšem už tato první úroveň zajistí, že jeden adresář bude obsahovat $n/16$ souborů, jelikož algoritmus SHA1 používá ve výsledném řetězci hexadecimální znaky.

6.5 Hromadné nahrávání a analýza dat

Protože systém bude obsahovat netriviální množství souborů a stejně tak do něho budou další soubory vkládány, lze očekávat, že nebudou vkládány po jednotlivých souborech, ale v rámci procesu digitalizace výkresové dokumentace vznikne balík souborů, které bude potřeba do systému vložit.

V raných fázích tvorby systému bylo v jazyce Python vytvořeno několik prototypů programů pro indexaci a vkládání souborů do databáze. Tyto prototypy se ukázaly jako velmi efektivní, avšak tento přístup k naplňování databáze se ukázal jako nepoužitelný. Největším problémem byla nutnost je spouštět v pravidelných intervalech, což znamenalo, že uživatel, který do systému vkládal data, se nedozvěděl, jak vkládání dopadlo, dokud program neskončil. Poté nastal problém, jak uživateli v rozumné podobě sdělit, kde nastal problém. Velký problém tohoto řešení také byl, že uživatel musel mít přístup k souborovému systému datového skladu nebo aplikačního serveru, což je velké bezpečnostní riziko. Celé toto řešení tak zpomalilo vkládání nových výkresů, jelikož nutnost uživatele čekat na výsledek nebyla vyvážena rychlostí samotného vkládání.

Z tohoto důvodu byla do systému přidána podpora pro hromadné vkládání souborů přímo prostřednictvím webového rozhraní, která uživateli zajistí jednoduchý způsob vložení těchto souborů. Jako nejlepší řešení se po sérii testů ukázal systém Plupload¹⁹ od společnosti Moxiecode Systems AB, který umožňuje využití mnoha technologií na straně klientského prohlížeče jako Flash, Google Gears, HTML 5, Silverlight, BrowserPlus a HTML 4. Právě podpora mnoha různých technologií a s tím spojená snaha o zachování funkčnosti napříč těmito technologiemi nabízí výslednému systému variabilitu nasazení.

¹⁹<http://www.plupload.com>

Přes jedno rozhraní tak bude možné vkládat velké množství souborů ze stolního počítače, stejně jako vložit fotografii výrobku ke konkrétnímu výkresu z mobilního telefonu. Velmi důležitý je také fakt, že technologie jako Flash, Google Gears nebo Silverlight mohou být ve firemním prostředí, z bezpečnostních a jiných důvodů, zakázány. Jelikož je podpora HTML 5 mezi prohlížeči již dostatečně rozšířená²⁰, zajišťuje podpora této technologie funkčnost i v restriktivním firemním prostředí.

Hromadné vkládání souborů díky využití technologií AJAX a JSON umožňuje uživateli okamžitě vidět, s jakým výsledkem byl soubor nahrán a případně na to ihned reagovat. Při nahrávání se systém pokusí pomocí definovaných regulárních výrazů najít v názvu souboru známé označení výkresu a jeho název. Toto řešení se ukázalo jako dostatečné a funkční. V rámci tvorby analyzátoru souborů s výkresy bylo vyzkoušeno několik OCR softwarů, které by uživatele zbavily nutnosti pojmenovávat soubory, avšak žádný z nich nebyl schopen s výkresy pracovat. Nasazení OCR by znamenalo značnou časovou úsporu, avšak vzhledem k chybovosti výstupů by musela být zajištěna vysoká spolehlivost a přesnost takového řešení. Implementace OCR je každopádně předmětem dalšího rozšiřování systému.

Po nahrání souboru a jeho analýze je rozlišováno několik stavů:

Výkres nebyl ze souboru identifikován - pokud nebyl výkres identifikován, je soubor uložen do dočasného adresáře, aby jej uživatel nebyl nucen znovu do systému vkládat, a je mu nabídnuta možnost ručně vyplnit informace o konkrétním výkrese

Výkres byl identifikován, ale neexistuje a soubor není vložen - v tomto případě se tedy jedná o zcela nový výkres. Systém tedy vytvoří výkres a záznam o vloženém souboru s vazbou na nově vytvořený výkres. Na tento stav nemusí uživatel nijak reagovat a systém jej pouze informuje o výsledku, což je vytvoření výkresu a přiřazení souboru.

Výkres byl identifikován a existuje, ale soubor není vložen - v tomto případě už výkres v databázi existuje a soubor má unikátní kontrolní součet. Je tedy uložen do datového skladu a je vytvořen záznam s vazbou na již existující výkres. Stejně jako v předchozím případě, ani na tento stav nemusí uživatel nijak reagovat. Pokud nahrává nový soubor, je s největší pravděpodobností přiřazení souboru k výkresu žádanou akcí.

Výkres byl identifikován, existuje a soubor je vložen - v tomto případě se uživatel snaží vložit soubor, z jeho názvu byl korektně identifikován již existující výkres a kontrolní součet souboru se shoduje se souborem, který je již v databázi vložen. Systém tak zamítne vložení souboru a uživatele o tom informuje. Na tento stav není dovoleno uživateli nijak reagovat.

²⁰<http://html5test.com/results.html>

7 Další rozvoj IS

Přestože systém disponuje všemi nezbytnými funkcemi pro svou funkčnost, je zde stále prostor pro rozšíření. Systém byl vyvíjen ve spolupráci se společností vlastnící archiv výkresové dokumentace a provádějící jeho digitalizaci. I přes úvodní analýzu bylo po nasazení do provozu identifikováno několik funkcí, které by mohly systém obohatit. Vývoj systému se rozhodně nezastavil.

7.1 Administrovatelné regulární výrazy pro identifikaci souborů

Současná podoba systému obsahuje několik předdefinovaných běžně používaných regulárních výrazů určujících tvar názvu souboru, které jsou využívány pro identifikaci čísla výkresu a jeho názvu. Tyto regulární výkresy vycházejí z potřeb jedné konkrétní společnosti a i přesto, že jejich přítomnost by dalším společnostem nevadila, s velkou pravděpodobností se objeví nový předpis pro název souboru a bude třeba vytvořit nový regulární výraz. Po vytvoření a odzkoušení regulárního výrazu je v současném systému nutné upravit zdrojový kód aplikace, což může být velmi nebezpečné a ne každá společnost zaměstnává vlastního programátora. Řešením by bylo vytvoření administračního rozhraní pro možnost vložení těchto regulárních výrazů a systém by je pak dynamicky používal.

7.2 OCR

Možnost nasazení OCR byla testována už v raných fázích tvorby systému avšak neúspěšně. Tyto snahy by však neměly zaniknout a systému by velmi prospěla možnost rozpoznání názvu výkresu a jeho čísla přímo ze souboru bez nutnosti zkoumat pouze název.

7.3 Podpora více datových skladů

Pro větší nasazení systému nebo pro nasazení do společností s geograficky oddělenými lokalitami s pomalým síťovým propojením by systém mohl podporovat uložení souborů a jejich replikaci mezi více datových skladů, které by tak mohly být blíže uživateli. Ten by poté z centrálního serveru získával pouze komprimované webové stránky a samotný soubor by již stahoval ze serveru připojeného po rychlé LAN síti.

7.4 Analyzátor využití jednotlivých souborů

Jak z důvodů možné prioritní replikace, tak z důvodů možného dlouhodobějšího cachování by bylo vhodné do systému dodat analyzátor využití jednotlivých výkresů a jejich souborů. Pro běžného uživatele je tato funkce nepotřebná, avšak pro administrátora systému a programátora by mohla být potřebná.

7.5 Identifikace verze DWG

Binární soubory DWG jsou vytvářeny v různých verzích CAD softwarů, kdy každá může do souboru zanechat něco vlastního a obecně nelze korektně otevřít soubory vytvořené v novější verzi CAD systému než ve které je otevírána. Z toho důvodu by bylo vhodné přidat k jednotlivým souborům indikaci verze DWG souboru, popřípadě i software, který soubor vytvořil.

7.6 Bezpečnější přihlašování

V systému je přihlášení prováděno pomocí formuláře, jehož data jsou následně přenášena nezabezpečeným HTTP protokolem. Je tak možné zachytit data putující k serveru a získat z nich login a heslo uživatele. Řešením tohoto bezpečnostního problému by bylo využití šifrovaného HTTPS nebo zasílání už pouze hash hesla, případně hash hesla s využitím jednorázového klíče.

7.7 Implementace NTLM

Pro nasazení ve Windows prostředí by bylo vhodné implementovat podporu NTLM²¹, což by uživatelům umožnilo automatické přihlášení do systému díky prvotnímu ověření vůči doménovému řadiči a náležitosti uživatele do domény.

7.8 Virtuální systém

V dnešní době, kdy se i v malých a středních firmách začíná rozmáhat virtualizace se jeví jako velmi dobré řešení vytvoření obrazu operačního systému včetně vytvořeného informačního systému a všech přidružených aplikací, který by dovolil velmi rychlé nasazení do stávající infrastruktury společnosti bez nutno další nákladů na pořízení fyzického serveru.

²¹<http://davenport.sourceforge.net/ntlm.html>

8 Závěr

V této bakalářské práci jsem rozebral problematiku digitalizace staré výkresové dokumentace do digitálně zpracovatelných formátů a vytváření moderního informačního systému s primárním zaměřením na uživatele demonstrované na potřebách konkrétní společnosti z praxe.

Výstupem této práce je analýza problematiky digitalizace výkresové dokumentace, analýza a návrh informačního systému a samotný informační systém jako velmi dobrý základ pro další rozvoj pomoci uživatelů v této oblasti.

Myslím, že práce je přínosná jak v teoretické, tak v praktické rovině, protože ukazuje reálné potřeby i přesto fungující společnosti. Na samotného uživatele byl brán ohled v průběhu celé práce a na výsledku se to zcela jistě projevilo. Finální podoba systému sice nepřekypuje líbivou grafikou, ale to uživatelé z výroby a konstrukce, kteří většinu své práce s počítačem tráví v programech jako AutoCAD, průmyslové řídicí systémy Sinumerik a CAM systémy, kde všude barva ikon nehraje nejmenší roli, nevyžadují a naopak oceňují zaměření na důležité informace.

Systém byl nasazen v praxi a díky jeho nenáročnosti na prostředky a připravenosti na začlenění do firemního prostředí se tak povedlo v rekordním čase, čímž předčil všechny systémy doposud nasazované ve společnosti MSV STUDÉNKA s.r.o.

Adam Surák

9 Reference

- [1] Gutmans, Andi; Saether Bakken, Stig; Rethans, Derick: *Mistrovství v PHP 5*, 2007. Nakladatelství Computer Press a.s. ISBN: 978-80-251-1519-0
- [2] Darie, Cristian; Brinzarea, Bogdan; Chereces-Tosa, Filip; Bucica, Mihai: *AJAX a PHP - tvoříme interaktivní webové aplikace PROFESIONÁLNĚ*, 2006. Zoner Press. ISBN: 80-86815-47-1
- [3] CAD studio a.s., *Jak propojit papír a CAD?* [online], [cit. 2011-02-15]. Dostupné z: <http://www.cadstudio.cz/paper.asp>
- [4] Petra Vávrová, Ústav chemické technologie restaurování památek, *Koroze a degradace papíru* [online], [cit. 2011-02-20]. Dostupné z: http://www.vscht.cz/met/stranky/vyuka/predmety/koroze_materialu_pro_restauratory/kadm/pdf/3_4.pdf
- [5] Petr Bouška, *Adresářové služby a LDAP* [online], 2007-09-14, [cit. 2011-02-24]. Dostupné z: <http://www.samuraj-cz.com/clanek/adresarove-sluzby-a-ldap/>